Chemical Engineering Science, vol. 65, no. 5, pp. 1694–1704, March 2010

Accelerated Computation of Cyclic Steady State for Simulated-Moving-Bed Processes

Hong-Mei Yao [1*], Moses O. Tadé [1], Yu-Chu Tian [2]

[1] Department of Chemical Engineering, Curtin University of Technology
GPO Box U1987, Perth, Western Australia 6845, Australia.
[2] School of Information Technology, Queensland University of Technology
GPO Box 2434, Brisbane, Queensland 4001, Australia

** Corresponding author: h.yao@curtin.edu.au, Tel: 61 8 9266 3677, Fax: 61 8 9266 2681*

ABSTRACT

Industrial applications of the simulated-moving-bed (SMB) chromatographic technology have brought an emergent demand to improve the SMB process operation for higher efficiency and better robustness. Improved process modelling and more-efficient model computation will pave a path to meet this demand. However, the SMB unit operation exhibits complex dynamics, leading to challenges in SMB process modelling and model computation. One of the significant problems is how to quickly obtain the steady state of an SMB process model as process metrics at the steady state are critical for process design and real-time control. The conventional computation method, which solves the process model cycle by cycle and takes the solution only when a cyclic steady state is reached after a certain number of switching, is computationally expensive. Adopting the concept of Quasi-Envelope (QE), this work treats the SMB operation as a pseudo-oscillatory process because of its large number of continuous switching. Then, an innovative QE computation scheme is developed to quickly obtain the steady state solution of an SMB model for an arbitrary initial condition. The QE computation scheme allows larger steps to be taken for predicting the slow change of the starting state within each switching. Incorporating with the wavelet-based technique, this scheme is demonstrated to be effective and efficient for an SMB sugar separation process. Moreover, investigations are also carried out on when the computation scheme should be activated and how the convergence of the scheme is affected by a variable stepsize.

Keywords: Simulated-moving-bed processes, Cyclic steady state, Dynamic simulation, Mathematical modelling, Model computation

1. INTRODUCTION

Simulated moving bed (SMB) chromatography is the technical realisation of a counter-current adsorption process. An SMB system consists of a certain number of chromatographic columns in series; while the countercurrent movement is achieved by sequentially switching the inlet and outlet ports one or more columns downwards in the direction of the liquid flow after a certain period of time. Major commodity applications of SMB separation processes can be found in petroleum, food, biotechnology, pharmaceutical, and fine chemical industries.

Generally, the SMB modelling approaches can be classified into the following two types:

- To describe the system in terms of an equivalent true moving bed (E-TMB) model; and
- To assemble the system by a series of dynamic SMB (D-SMB) models of single chromatographic columns under explicit consideration of cyclic switching operations.

The main difference between E-TMB and D-SMB modelling is their stationary regime to be considered. The E-TMB modelling represents the traditional concept of steady state operation while the modelling using D-SMB deals with the cyclic steady state explicitly.

An E-TMB model is represented by a set of ordinary differential equations (ODEs) and algebraic equations. When the cyclic port switching is neglected, the E-TMB model is simplified and thus can be solved very efficiently using existing numerical computation methods. It has been found that the steady-state solution of a detailed E-TMB model reproduces the solution of an SMB model reasonably well under certain circumstances (Hashimoto et al., 1983; Storti et al., 1988; Pais et al., 1998).

However, the E-TMB modelling is a rather severe idealisation of the simulated counter-current process. It is restricted to the case of three or more columns per zone with linear isotherms and without reaction (Dünnebier et al., 2000). In the case of a

three-section process, the E-TMB approach is of limited quantitative value (Chu & Hashim, 1995), and thus needs to be treated with caution. Comparative simulations (Strube & Schmidt-Traub, 1998) have also shown that SMB optimisation based on E-TMB modelling will lead to a miscalculated operating condition since E-TMB models do not consider the transient regime of the quasi-stationary, periodic steady state of SMB processes, and the upper and lower bounds of the retention time.

Earlier applications of the SMB technology were found in food and petrochemical industries in large separation units, which were operated under nearly linear behaviour of the equilibrium isotherm. Most of the recent applications are in the processes for fine chemicals and pharmaceutical products, and are operated under nonlinear adsorption behaviour and in a high purity range. Thus, SMB modelling through dynamic simulation is essential for more-accurate descriptions of the chromatographic processes and their dynamic behaviour. It is also crucial in development of control strategies as well as in analysis of the stability of the SMB processes to reduce malfunctions.

Since dynamic SMB modelling requires solving a set of partial differential equations (PDEs) and algebraic equations, it brings the difficulty in numerical computing and requires much more computational effort than that for solving E-TMB models. This is a deterrent for online applications. Over the past decade, limited simulation strategies have been investigated for SMB chromatographic systems (Nilchan & Pantelides, 1998; Kloppenburg & Gilles, 1999; Jiang et al., 2003; Platte et al., 2005; Lübke et al., 2007). Among these works, there are essentially two approaches to determine the steady state of a cyclic operation, the dynamic simulation approach and the direct prediction approach (Minceva et al., 2003). The former is implemented through successive substitution, while the latter solves the problem through embedding the periodic boundary conditions into the problem formulation. The difference between those two approaches lies in the way of iteration for the whole cycle or just one switching time, starting from the given initial conditions or using the cyclic steady state profile from previous simulation. These approaches mimic the start-up of the cycle accurately with a repeated procedure for a large number of cycles (more than 10 cycles, or over 100 switching), but can also be time consuming and computationally expensive.

Addressing the challenge in the determination of the steady state through dynamic simulation, this work considers the SMB cyclic operation as a highly oscillatory process, leading to a completely new approach for achieving the steady state in the model computation. A fast integration scheme is developed for SMB processes based on the concept of "Quasi-Envelope". Avoiding the detailed calculation of the transient equations, it provides an indirect prediction of the cyclic steady state via a pseudo smooth function. The advantage of the proposed scheme stems from the fact that for such a cyclic operation, stationary conditions prevail when the cycle time is much less than the characteristic process time, i.e., the time required for achieving the cyclic steady state.

2. PROBLEM DESCRIPTION

2.1 The Concept of "Quasi-Envelope"

The original idea of "Quasi-Envelope" is related to solving differential equations with highly oscillatory solutions. It can be traced back to as early as 1957, when astronomers firstly introduced the so-called multi-revolution methods for calculating the orbits of artificial satellites. The more formal "Quasi-Envelope" method was proposed by Petzold (1981), who was inspired by the observation that when a nearly periodic function is sampled at multiples of the period (or near-period) of the oscillation, the resulting sequence of points changes slowly. The method aimed at finding the slow change behaviour of the solution without having to closely follow the oscillation details. In the following, we will firstly present how the problem is formulated in mathematical descriptions.

Consider an initial value problem

$$y'(t) = f(y(t),t),\ y(0) = y_0 \qquad (1)$$

where $y(t)$ is a periodic or near periodic function with a period of T, t is time, and y_0 is the initial condition. A typical solution of this problem in many applications is a sinusoidal wave, as shown in Figure 1.

Ignoring the details of every oscillation in the solution of $y(t)$, we consider the peak of each period instead. When the oscillation is very fast, a smooth curve, which passes through these multiples of period, can be defined. This is the so-called quasi-envelope. Using $z(t)$ to represent the quasi-envelope of the original function $y(t)$, according to the definition, we have the following two properties (Petzold, 1981):

Property 1: The state on $z(t)$ can be found at discrete points, one period time apart. Those points should agree with the solution to the original problem, especially, $z(0) = y(0)$, and in general, $z(nT) = y(nT)$, where n is an integer.

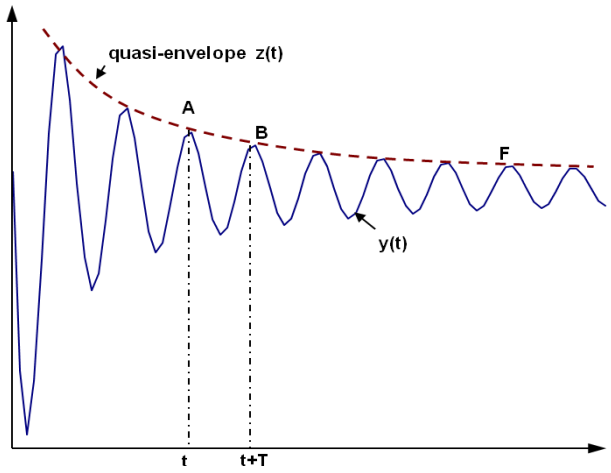


Figure 1 Sinusoidal wave and its quasi-envelope. y(t) is the original function and z(t) is a fictitious curve passing through the peaks of oscillations.

Since this is a slowly varying function compared to oscillation, it can be followed in a large step H to approximate the value at the interval $H >> T$. More precisely, the property of the quasi-envelope determines that the value at multiples of the period agrees with the solution to the original equation. Starting from any of the points, the point-wise solution to the original problem can be recovered through integration using that particular point as initial condition. For example, in Figure 1, if we know $z(t)$ at point *A*, then the curve underneath *A*-*B* can be found through one period integration to *y*(*t*).

Property 2: $z(t)$ *is a fictitious curve, thus the derivatives of* z *with respect to time cannot be found through mathematical operation. However, the derivatives of* z *is a must for numerical computations. Since* $y(t)$ *is nearly periodic, the values of* $z(nT)$ *should change slowly. The derivatives of* z *with respect to time may be approximated by the change of function* $z(t)$ *during one period* T*, that is;*

$$z'(t) \approx g(z) = \frac{z(t+T) - z(t)}{T} \qquad (2)$$

Petzold's algorithm (Petzold, 1981) is analogous to Gear's multivalue or multistep algorithm but uses the difference quotients in Equation (2) instead of derivatives. Multivalue methods require several pieces of information about the dependent variable at time t_{n-1} to compute the equivalent pieces of information at t_n. If the values of the dependent variable and its derivative at k different time points $t_{n-1}, t_{n-2}, \cdots, t_{n-k}$, are used in the computation, the corresponding methods are commonly called Multistep methods; in this case, k-step method (Gear, 1971). This method consists of two processes, the prediction and the correction.

The most popular multistep combination is the explicit Adams-Bashforth predictor and the implicit Adams-Moulton corrector (Davis, 1984), as will be briefly discussed below. Denote H as the step size of the integration for function $z(t)$, the multistep integration formulae can be represented by the following equations (Davis, 1984) :

$$z_{n,(0)} = \sum_{j=1}^{k} [\alpha_j(r) z_{n-j} + H\beta_j(r) g_{n-j}] \qquad \text{(Adams-Bashforth prediction)} \qquad (3)$$

where $r = T / H$, α and β are two coefficients.

The prediction process does not make use of the original differential equation, so a correction process is imposed to correct the approximate value. The amount by which $z_{n,(0)}$ does not satisfy the real value is:

$$G(z_n) = g(z_n) - g_n \qquad (4)$$

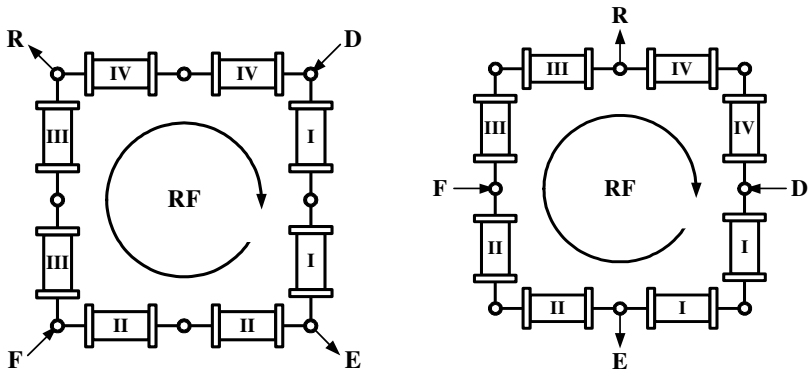
Thus, we have the following Adams-Moulton correction formula (Davis, 1984),

$$z_{n,(m+1)} = \sum_{j=1}^{k} [\alpha_j^*(r) z_{n-j} + H\beta_j^*(r) g_{n-j}] + H\beta_0^*(r) G(z_{n,(m)}) \qquad (5)$$

Starting from the predictor in Equation (3), Equation (5) is calculated to correct the result. Then, the process is repeated for $m = 1,2\cdots$ iterations until there is no further change in $Z_{n,(m+1)}$. Because $z(t)$ is discretised at nT, where the quasi-envelope and original function are crossed, the state of z at this point also agrees with the state of y. Thus, $z(t)$ provides a starting point for the integration of $y(t)$. If H is chosen as several times of T, say, from *B* directly to *F* in Figure 1, the above calculation will give the state of $z(t)$ several periods away. Using this state as initial condition, we can reconstruct $y(t)$ for the next full period or more.

2.2 "Quasi-Envelope" for SMB Model Computation

At first, the principle of SMB operation is briefly explained below to help understand the implication of the computational algorithm to be proposed. The mechanism is illustrated in Figure 2.



(a) Current switching period (b) Next switching period

Figure 2 An SMB unit with 2 columns in each of the 4 sections
(RF: direction of fluid flow and port switching)
F: Feeding; D: Desorbent; R: Raffinate; E: Extract

The system consists of several columns connected in series to form a circular loop. According to the functional requirements, it can be divided into 3 or 4 sections. But the number of columns in each section can be different. Four ports are opened for feeding of mixture, desorbent and drawing of fluids. While the fluids are circulating inside the columns, the feed mixture, desorbent, extract, and raffinate are continuously allowed to enter or leave the system from the openings (Figure 2a). Then, successively, at a designed time instant, these four ports are moved in the direction of the circular flow for one full column (see Figure 2b). As a result, it seems as if the adsorbent (solid phase) moves in the opposite direction to the flow of the fluids. Therefore, mixed components can be removed in a continuous manner from its respective withdrawal point.

A well-known application of this principle is the SORBEX system, where a rotatory valve controls the motion of the ports along the columns. The same result can be achieved by connecting each section of the column with many on-off valves; each of these is in turn a feed point or draw off point or simple connection depending on the particular role of the section at a particular time. Typically, four-section cascade SMB is the usual process scheme adopted for commercial applications and academic research.

The way of the above mentioned operation implies that SMB chromatography is a cyclic process. The stationary regime of the process is cyclic steady-state (CCS), which means at that steady-state, an identical transient behaviour takes place during each period. The CSS is practically reached after a certain number of valve switches, but the system states are still varying over time because of the periodic movement of the inlet and outlet ports along the columns. A typical time profile of the concentration distribution along the column length is depicted in Figure 3 for an eight-column binary separation system.

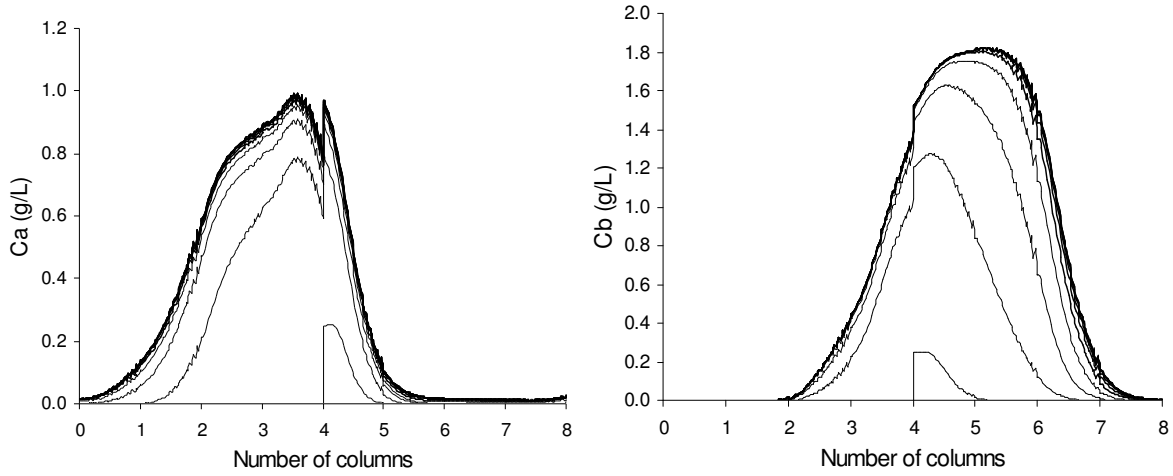


Figure 3 Propagation of wave in a binary separation SMB system with configuration of 2-2-2-2. Each line represents the distribution of concentration at certain time instant in a switch. They are gradually convergent to a cyclic steady state.

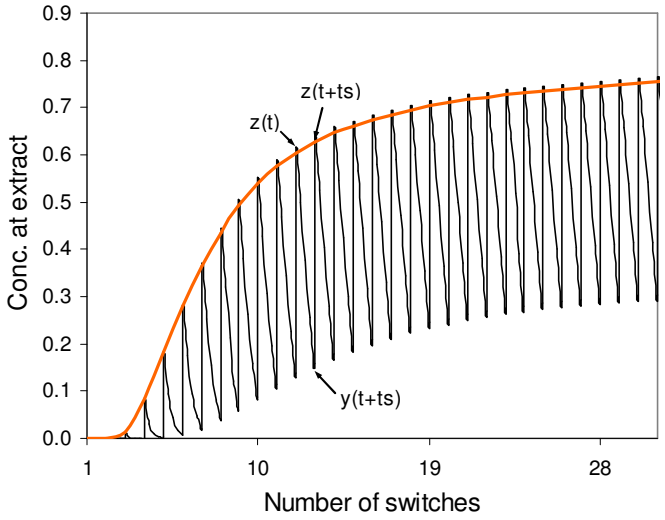


Figure 4 Time profile of extraction y(t). Each oscillation represents a switching period, starting from the top line and finishing at the bottom. The proposed underlying quasi-envelope z(t) is a slowly-varying smooth curve.

It might still be difficult to relate this process to oscillatory functions illustrated in Figure 1. Since this is a multivariable problem, let us take the concentration from the extraction port as an example. Figure 4 is the time profile of the average concentration for one of the components being separated.

For an SMB to reach steady state, it takes at least 10-12 cycles (for an 8-column SMB, it is 80-96 switching). So the oscillation is highly condensed in terms of the process time constant. Thus, the SMB operation can be treated as a pseudo-oscillatory process. With this in mind, the implication of Petzold's idea becomes obvious. Instead of using traditional switch by switch (or cycle by cycle) integration, we can take advantage of quasi-envelope function as an express train to obtain the steady state faster by ignoring the details of some of the switching periods. Particularly, this is a process with known constant period (the switching time), which makes the application of Petzold's idea to SMB simulation straight forward.

3. THE ALGORITHM AND IMPLEMENTATION

The multistep method represented by Equations (3) and (5) uses current and previous points to predict the value at the next point. For stiff problems, Newton iteration is usually an essential part of a multistep stiff solver. However, multistep methods as we described above suffer from two serious difficulties:

(1) Since the formulae require results from equally spaced steps, adjusting the stepsize is difficult; and
(2) The starting and stopping problem also exist. For starting, we need not only the initial values but also several previous steps. For stopping, equal steps are unlikely to land directly on the desired termination point.

To overcome these difficulties, especially for a variable step size, which is the key for this quasi-envelope method, Petzold has re-written the algorithm in the generalised Adams formula using multivalue methods. We will present the equations in the following section directly. Interested readers should refer to Petzold's original work (Petzold, 1981) and Gear's book (1971).

3.1 Generalised Adams Formula from Multivalue Method

For multivalue methods, the basic data available to the integrator are the fist few terms of the Taylor series expansion of the solution at the current point t_n.

$$y_{n+1} = y_n + Hy'_n + \frac{H^2}{2} y''_n + \cdots + \frac{H^k}{k!} y_n^{(k)}$$

This is a contrast to the multistep methods, where the data are the values of the solution at $t_{n-1}, t_{n-2}, \cdots, t_{n-k}$, as shown in Equations (3) and (5). The use of the generalised Adams formula from multivalue method would allow easy adaptation to a variable stepsize.

Define a vector $\mathbf{w}_n$, so that $\mathbf{w}_n$ is a matrix of several pieces of information from t_n step only,

$$\mathbf{w}_n = [z_n, Hg_n, \frac{H^2}{2} g'_n \cdots, \frac{H^k}{k!} g_n^{(k-1)}]^T$$

Also, $\mathbf{w}'_n$ is defined as containing the information from previous several steps:

$$\mathbf{w}'_n = [z_n, Hg_n, Hg_{n-1} \cdots, Hg_{n-k-1}]^T$$

Where $g_n = \dfrac{z(t_n + T) - z(t_n)}{T}$

Then, $\mathbf{w}_n$ and $\mathbf{w}'_n$ can be related by the linear transformation:

$$\mathbf{w}_n = \mathbf{S}\mathbf{w}'_n \qquad (6)$$

The value of $\mathbf{S}$ is determined by the integration order. Petzold allows the integration order to vary from 1 to 11. For our purpose, the first-order method will not be used and much higher orders are also not applied considering the stability of the algorithm. We will conduct our investigation on order 2.

For the second-order (*k=3*) multivalue method, it follows that

$$z_n = z_n, \qquad g_n = g_n, \qquad g_{n-1} = g_n - Hg'_n$$

So, $\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1/2 & -1/2 \end{bmatrix}$

Then, the generalised Adams prediction and correction method is given by

$$\mathbf{w}_{n,(0)} = \mathbf{A}\mathbf{C}(R_n)\mathbf{w}_{n-1} \qquad (7)$$

$$\mathbf{w}_{n,(m+1)} = \mathbf{w}_{n,(m)} + \mathbf{B}H_n\mathbf{G}(z_{n,(m)}) \tag{8}$$

with $R_n = H_n / H_{n-1}$ being the stepsize ratio for the consecutive integration step.

$\mathbf{C}(R_n) = diag[1, R_n, R_n^2, \cdots, R_n^k]$;

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1-r & 1-\frac{3}{2}r+\frac{1}{2}r^2 & \cdots & \cdots \\ & 1 & 2 & 3 & & \\ & & 1 & 3 & & \\ & & & 1 & & \\ & & & & \ddots & \\ & & & & & 1 \end{bmatrix};$$

$\mathbf{B} = \mathbf{Sc}$;

$\mathbf{c} = [\beta_0^*(r), 1, 0, \cdots, 0]^T$;

For second-order ($k=3$), $\beta_0^*(r) = 1 - r$; and $\mathbf{B} = \begin{bmatrix} \beta_0^*(r) \\ 1 \\ 1/2 \end{bmatrix}$

$\mathbf{G}(z_{n,(m)})$ is defined by Equation (6.4), in vector:

$$\mathbf{G}(z_n) = \mathbf{g}(z_n) - \mathbf{g}_n \tag{9}$$

where $\mathbf{g}_n$ is the predicted value included in the vector $\mathbf{w}_n$.

In classical numerical methods, $\mathbf{g}(z_n)$ is $\mathbf{z}_n'$, which can be calculated if the function of $z'(t)$ is given. In this Quasi-Envelope technique, $g(z_n)$ is defined by

$$g(z_n) = \frac{z(t_n + T) - z(t_n)}{T}$$

$z(t_n + T)$ will be obtained through one cycle integrating on the original function using predicted z_n, so that,

$$\mathbf{g}(\mathbf{z}_n) = \frac{\mathbf{y}(t_n + T) - \mathbf{z}(t_n)}{T} \tag{10}$$

It can be seen that Equation (7) allows a variable stepsize. Changing the stepsize by a ratio R_n corresponds to multiplying $\mathbf{w}_{n-1}$ by the matrix $\mathbf{C}(R_n)$.

3.2 Periodic Integration Procedure

The time integration is conducted on the ODEs transformed from PDEs. The complete solution scheme is proposed in Figure 5.

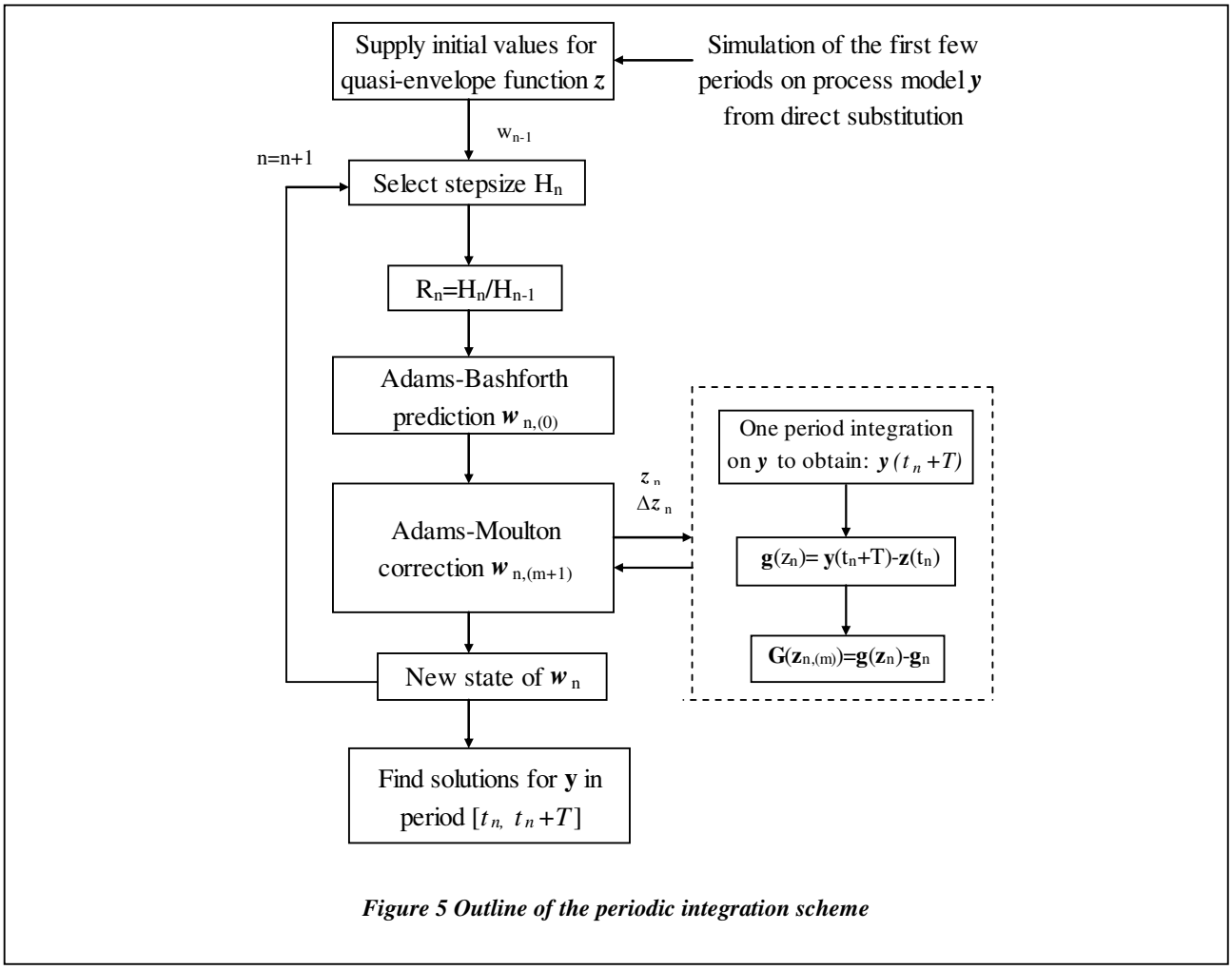


Figure 5 Outline of the periodic integration scheme

This scheme involves two integration operations, one for the original function $y(t)$ and the other for the quasi-envelope function $z(t)$. The multivalue method given in the previous section applies to the integration of $z(t)$, called "outer integration". However, during the approximation of $z(t)$, the corrector also requires a one cycle evaluation of $y(t)$, named "inner integration".

The implementation of the above algorithm is described in the following procedure:

Step 1: The integration routine starts by taking several steps with step size equal to one switching time. Cares must be taken to ensure that the initial evaluations should provide sufficiently accurate information for $z(t)$ to start the integration.

$z(0) = y(0);$ $\quad z(1) = y(H);$

$g(0) = y(T) - y(0);$ $\quad g(1) = y(H+T) - y(H);$

$g'(0) = 0;$ $\quad g'(1) = \dfrac{g(1) - g(0)}{H};$

$g''(0) = 0;$ $\quad g''(1) = \dfrac{3!}{H^3}[z(1) - z(0) - Hg(1) + \dfrac{H^2}{2} g'(1)]$

$g''(1)$ is derived from third order Taylor expansion.

Step 2: Predict $\mathbf{w}_{n,(0)}$ using Equation (7)

Step 3: Using Newton iteration to calculate $\mathbf{w}_{n,(m+1)}$ from Equation (8) until it meets the convergence requirement.

One of the correction steps is:

$$\mathbf{w}_{n,(m+1)} = \mathbf{w}_{n,(m)} + (H_n / T)\mathbf{B}[\mathbf{I} - (H_n / T)\beta_0^*(r)\mathbf{\Phi}]^{-1}\mathbf{G}(z_{n,(m)}) \tag{11}$$

Step 4: Choose next stepsize and repeat from Step 2.

This procedure is very standard for a multistep numerical integration. Potential problems are in the evaluation of the Jacobian matrix $\mathbf{\Phi}$ for the Newton algorithm. The construction of the Jacobian requires the call for inner integration. The computation time is the dimension of $\mathbf{w}$ multiplied the time used for the evaluation of a single switch. For SMB models, the dimension of $\mathbf{w}$ can be enormous. This is one of the problems we addressed in this investigation.

4. APPLICATIONS TO SMB CYCLIC PROCESSES

4.1 Dynamic SMB Modelling

A dynamic SMB model consists of two main parts: a single chromatographic column model and a node balance model. The model has a set of PDEs for mass balance over the column, ODEs for parabolic intraparticle concentration profile, and algebraic equations for equilibrium isotherms and node mass balances. The level of mathematical difficulty encountered depends on the nature of the equilibrium relationship, the concentration level, and the choice of flow models (Ruthevn, 1984).

Among the column models for chromatography, the Transport-Dispersive Linear Driving Force (LDF) models have been widely used in modelling many adsorption processes due to their simplicity and good agreement with experimental results (Ching, 1997). LDF is the application of the first Fick's law of mass transport and is an equation of equilibrium isotherm. It offers a realistic representation of industrial processes and is shown to be a good compromise between accurate and efficient solutions of these models (Biegler et al., 2004). In this study, we will use the following typical expression of Transport-Dispersive-LDF model to describe the kinetics of each column.

$$\frac{\partial C_{i,j}}{\partial t} + u_j \frac{\partial C_{i,j}}{\partial x} = D_{ax,j} \frac{\partial^2 C_{i,j}}{\partial x^2} - \frac{1-\varepsilon_b}{\varepsilon_b} k_{eff,i}(q_{i,j}^* - q_{i,j}) \tag{12}$$

$$\frac{\partial q_{i,j}}{\partial t} = k_{eff,i}(q_{i,j}{}^* - q_{i,j}) \tag{13}$$

To complete the dynamic modelling system, initial conditions and boundary conditions are also essential. These are summarised below.

Initial conditions:

Initial conditions describe the status of the column at the beginning of the switching

$$C_{i,j}^{[k]}(0,x) = C_{i,j}^{[k-1]}(t_s, x) \tag{14}$$

k: number of switching; i = component A or B; j = column number

Boundary conditions

There are two boundary conditions, one at the column inlet and the other at the column outlet.

The condition for $x = 0$ at $t > 0$ is: $$\frac{\partial C_{i,j}}{\partial x} = \frac{u_j}{D_{ax,j}}(C_{i,j} - C_{i,j}^{in}) \tag{15}$$

The condition for $x = L$ at $t > 0$ is: $$\frac{\partial C_{i,j}}{\partial x} = 0 \qquad (16)$$

where each $C_{i,j}^{in}$ is subject to the material balances at the column conjunctions as listed in Table 1.

***Table** 1. Node Model*

	Flow Rate Balance	**Composition Balance**
Desorbent node (eluent)	$Q_I = Q_{IV} + Q_D$	$C_{i,I}^{in} Q_I = C_{i,IV}^{out} Q_{IV} + C_{i,D} Q_D$
Extract draw-off node	$Q_{II} = Q_I - Q_E$	$C_{i,E} = C_{i,II}^{in} = C_{i,I}^{out}$
Feed node	$Q_{III} = Q_{II} + Q_F$	$C_{i,III}^{in} Q_{III} = C_{i,II}^{out} Q_{II} + C_{i,F} Q_F$
Raffinate draw-off node	$Q_{IV} = Q_{III} - Q_R$	$C_{i,R} = C_{i,IV}^{in} = C_{i,III}^{out}$
Other nodes	**Equal flow rates for the columns in the same zone**	$C_{i,j}^{in} = C_{i,j-1}^{out}$

The separation of fructose-glucose in de-ionized water used in Beste *et al*. (2000) has been chosen as our case for study. The system has 8 columns with the configuration of 2:2:2:2 as shown in Figure 2. The operating conditions and model parameters are summarised in Table 2. The model consists of 16 PDEs, 16 ODEs and 20 algebraic equations connecting all the variables together.

***Table 2** Parameters and operating conditions for fructose-glucose separation*

Symbol	**Value**	**Symbol**	**Value (A-fructose)**	**Value (B-glucose)**
L (cm)	52.07	$k_{eff,i}$ (min^{-1})	0.72	0.9
D (cm)	2.6	$C_{i,feed}$ (g/L)	363	322
ε_b	0.41448	$q_A^* = 0.675 C_A$ $q_B^* = 0.32 C_B + 0.000457 C_A C_B$		
t_s (min)	16.39			
Q_F (ml/min)	1.67			
	Zone I	**Zone II**	**Zone III**	**Zone IV**
Q (ml/min)	15.89	11.0	12.67	9.1
D_{ax} (cm^2/min)	1.105	0.765	0.881	0.633

For the inner integrator block, we adopt the wavelet-collocation based PDE solver we recently developed (Yao et al. 2008) for spatial discretisation using *J = 4* (*M = 1*), and the Alexander semi-implicit method (3rd order Runge-Kutta) for time integration. The outer integration is conducted using the generalised Adams prediction and correction methods outlined in this paper.

4.2 Determining the Initial Point for Quasi-Envelope

As pointed out previously, the initial state provided to the quasi-envelope is very important to ensure sufficient accuracy. Determining initial state can be done by taking several steps of detailed integration with step size equal to one switching time. Since the step size for quasi-envelope is set to be one cycle (8T), we will firstly conduct detailed integration for one cycle by direct substitution. After the 8^{th} switch, the feeding port sits back to the original position. This procedure produces the concentration profile at extract port as presented in Figure 6.

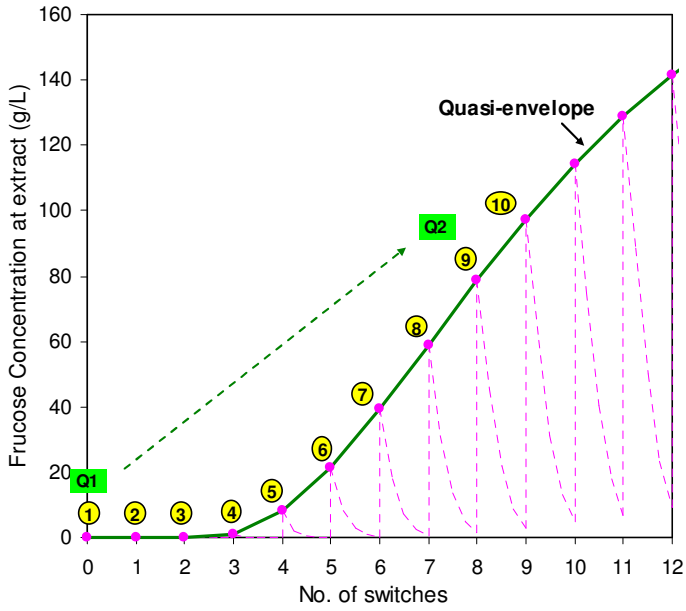


Figure 6 Concentration profile at extract for the start-up (1 to 10: the starting state of the real process within each switch; Q1 and Q2: the first two points for quasi-envelope, which are 8T apart).

For the second-order multivalue method, it is necessary to provide quasi-envelope function $z(t)$ the starting state Q1 and Q2. These are obtained by conducting 9 steps of inner integration. Because we have assumed constant time step for quasi-envelope, the next point will be another 8 switches away. Following the steps outlined in the previous section, approximation of $z(t)$ will start.

4.3 QE Prediction vs Successive Substitutions

It can be seen from the outer integration that during the correction step, the evaluation of Jacobian matrix $\mathbf{\Phi}$ for the Newton iteration requires call for inner integration. The inner integration should be calculated many times because the dimension of $\mathbf{w}$ is significantly high for an 8-column binary separation process, which is almost the basic configuration for SMB systems. A larger number of columns are expected for real SMB practice. One of the solutions to the problem is to evaluate $\mathbf{\Phi}$ initially and thereafter only if the corrector iteration convergence is too slow (Toftegard & Jorgensen, 1989). If $\mathbf{\Phi}$ must be re-evaluated quite often, a Broyden update may be used (Toftegard & Jorgensen, 1989).

In our study, we will evaluate the Jacobian matrix based on the point where the maximum prediction error occurs. This will lift the call of inner integration out of the construction of Jacobian matrix loop. So the number of using *inner* is equal to the number of Newton Iterations in the Adams correction step, which will significantly release the computational effort. Simulation results are given in Figures 7 and 8 using the second-order multivalue method.

The discrete $z(t)$ is evenly distributed and the approximation is gradually getting very close to the one obtained from successive substitution. The successful reconstruction of concentration distribution at steady state proves that predicted starting state from this method can be safely used.

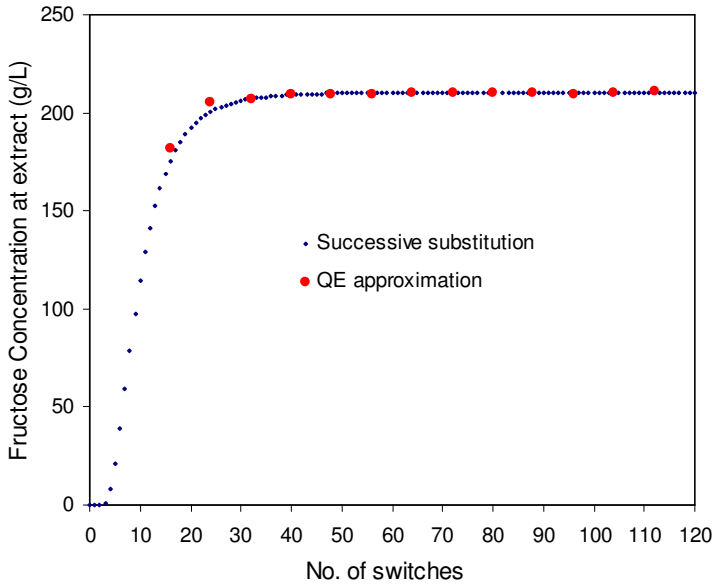


Figure 7 Comparison of calculated concentration state at the beginning of each switch from the methods of successive substitution and QE approximation.

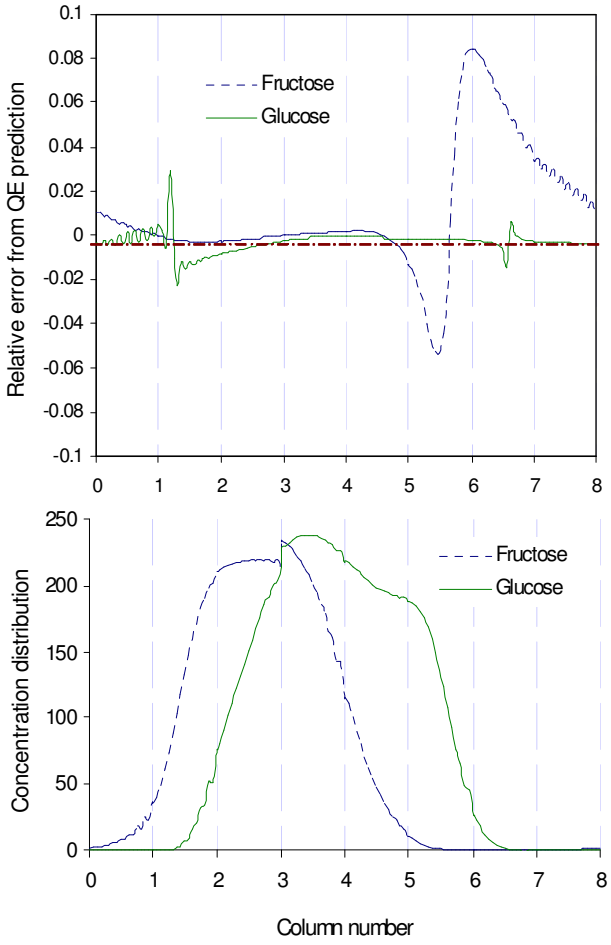


Figure 8 Steady state concentration and its relative error distributions from QE prediction

Recall that the sub-function *inner* runs evaluation of $y(t)$ for one switch. Thus, apart from accuracy, the Newton iteration, which is required in the corrector, can be an important criterion for this proposed method. For this trial, the Newton iteration required during correction step is depicted in Figure 9. Although the prediction accuracy of the starting state is promising (as shown in Figure 7), the correction step demands too many iterations in the first few cycles.

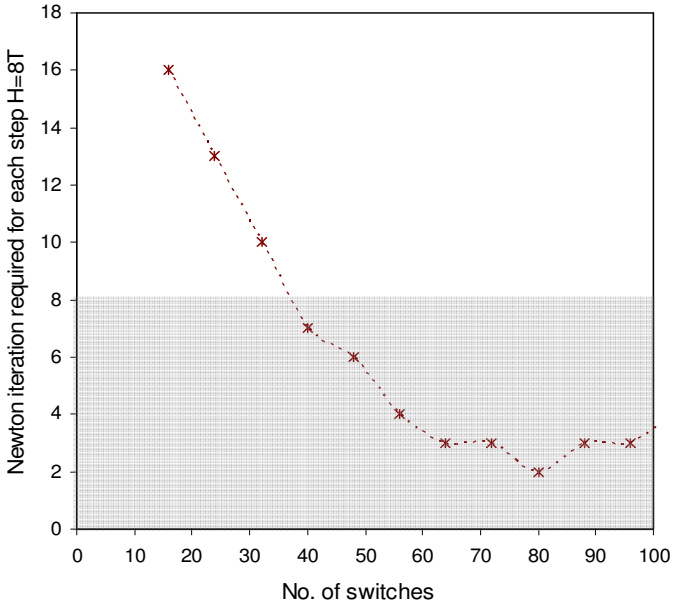


Figure 9 Newton iterations required for second order multivalue method.

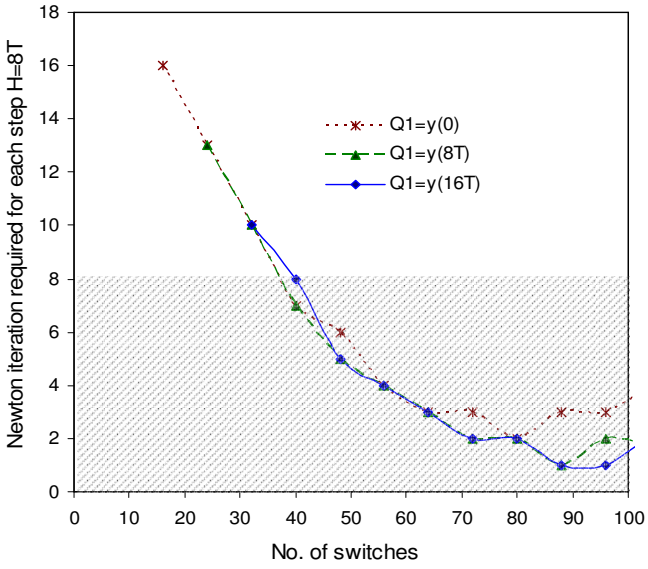


Figure 10 Correction step required during the prediction of $z(t)$ under different starting point, with septsize of H=8T. For $Q1 = y(0)$, the first predicted QE point is at the 16th switch. For $Q1 = y(8T)$, the first prediction of QE is at the start of the 24th switch, and so on

The advantage of using the quasi-envelope is obvious since an SMB operation involves many more switching, i.e. the process is stretched much longer. For ultimate use of this method at the start-up, further investigation has been carried out from the following aspects:

- *When should the QE line be activated, e.g. choosing the first point of Q1?*
- *How does a variable stepsize affect the performance?*

5. FURTHER INVESTIGATION INTO THE ALGORITHM

5.1 Activating the Quasi-Envelope Computation

We still use the previous constant integration stepsize, $H = 8T$. The first point of $z(t)$ is moved forward for one cycle (8T). As a result, we are gradually restricting the number of correction steps into the desired zone, as shown in Figure 10. Table 3 is also given to help understand this mechanism.

Table 3 *Prediction result from different Q1 location (H=8T).*

No. switch	Required correction steps			Predicted starting state			Starting state from Successive substitution
	Q1=y(0)	Q1=y(8T)	Q1=y(16T)	Q1=y(0)	Q1=y(8T)	Q1=y(16T)	
0							0.00
8							78.70
16	16			181.55			175.64
24	13	13		205.62	202.66		200.53
32	10	10	10	206.77	207.35	207.44	207.13
40	7	7	8	209.39	208.88	208.94	209.19
48	6	5	5	209.70	209.59	209.56	209.90
56	4	4	4	209.80	209.86	209.88	210.17
64	3	3	3	210.27	210.05	210.09	210.27
72	3	2	2	210.52	210.28	210.22	210.32
80	2	2	2	210.28	210.44	210.28	210.33
88	3	1	1	209.87	210.44	210.25	210.34
96	3	2	1	209.80	210.18	210.15	210.34
Switch/Step Ratio	88/70	80/49	72/36				

Here, the Switch/Step ratio “80/49” stands for a total of 49 correction steps in the outer integration to predict 80 switches in the real process operation. The higher this ratio the better the performance is. It is observed that the call for inner integration is significantly reduced when QE is activated after the first few cycles.

To evaluate the accuracy of this computation scheme, let us compare the predicted starting state with the calculated starting state from successive substation using the absolute relative error as criterion. The comparison is shown in Figure 11. The absolute relative error for QE prediction can be kept under 0.2% if the quasi-envelope is activated after the first two cycles.

Similar results can be obtained with constant stepsize of $H = 6T$ (Figure 12) and $H = 10T$ (Figure 13).

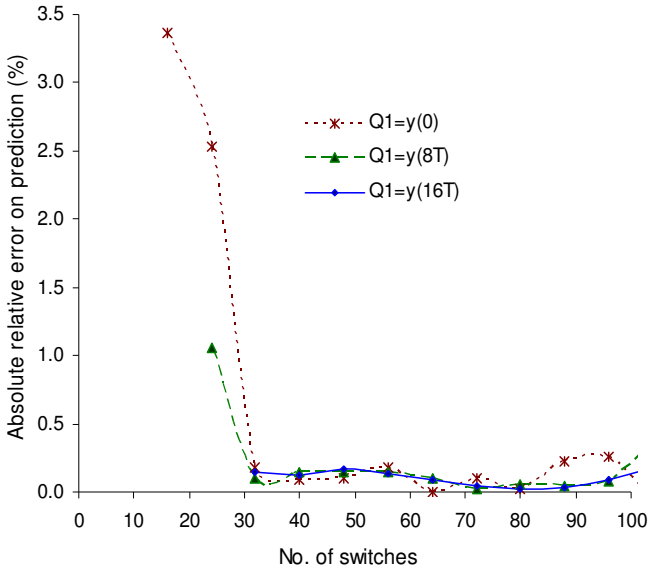
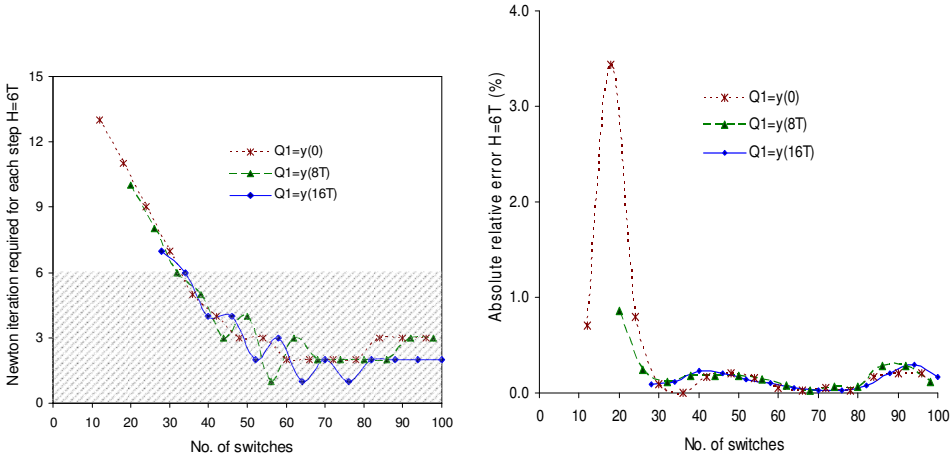
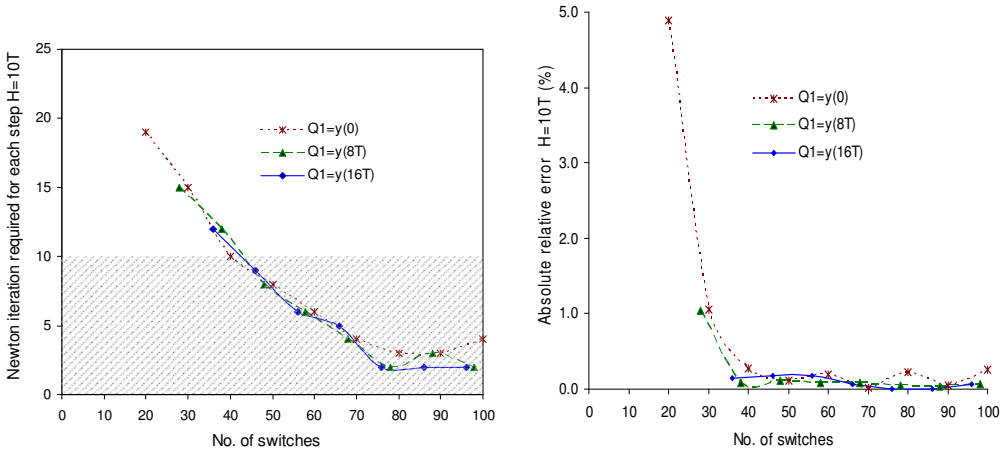


Figure 11 Absolute relative error for QE prediction with H=8T.



(a) ***(b)***

Figure 12 QE prediction with $H = 6T$ (a) correction steps required; (b) Absolute relative error.



(a) *(b)*

Figure 13 QE prediction with $H = 10T$ (a) correction steps required; (b) Absolute relative error.

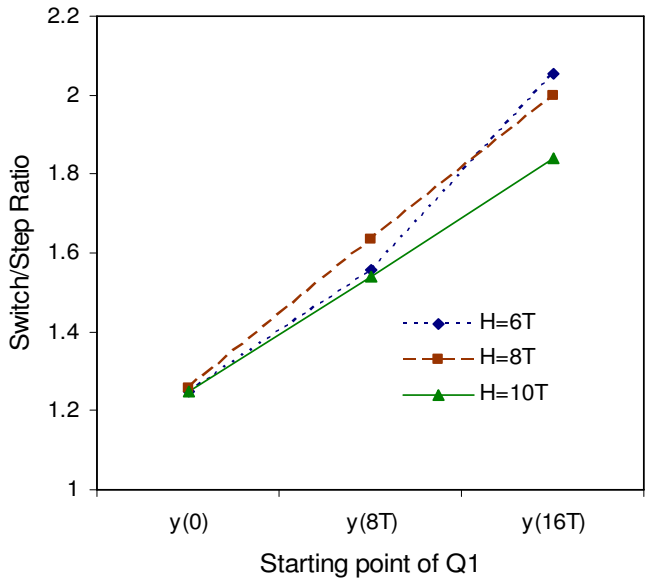


Figure 14 The Switch/Step ratio from QE integration

The impact of the step size and starting point can be seen from Figure 14, where their switch/step ratios are drawn against different starting points. The ratio is based on the minimum switch required to reach a steady state.

From these trials, the following results can be obtained:

- If choosing $z(0) = y(0)$, the multivalue algorithm would require similar computational effort (total number of correction steps) to achieve the steady state prediction of $z(t)$, regardless of the integration stepsize.
- If using a constant stepsize, activating QE later (after the initial start-up period, i.e., 1-2 cycles) is helpful when the quasi-envelope displays steep change at the start-up period.
- The quasi-envelope method would be of great advantage to predict the long term SMB operating states.

5.2 Investigation into Variable Stepsize

As shown in Figure 4, the shape of the quasi-envelope which we approximated is rather an exponential function than a linear one. The derivation of this algorithm has taken into account a variable stepsize. It has been pointed out (Davis, 1984) that a variable stepsize can be useful for controlling the local truncation error and improving efficiency during the solution of a stiff problem. We will explore this possibility in our applications to see if it helps with the improvement of performance. Nevertheless, attentions will be paid to the cost of the computation.

There are two ways under consideration, the judicious choice of the stepsize and the adaptive determination of the stepsize.

For the latter, adaptive stepsize strategies rely on the local or global estimation error. It involves the choosing of the stepsize for one step, the adopted approximation algorithm and its accuracy order, definition of test criteria and an expression for the stepsize to use for the next step or to repeat the rejected step. No matter which method is used, the basic mechanism is to execute one step and to perform the test. If the test succeeds, the step is accepted; otherwise, it is rejected. However, during this execute-test procedure, extra work is inevitably required on top of the standard numerical computing method.

Alternatively, it is possible to specify analytically how the stepsize should be varied if one has the knowledge of the process being studied. This would eliminate the test procedure. However, before using any stepsize strategies, we conducted a few trials by changing stepsize with an increment of one T for each integration step, or with a pre-defined pattern set up as below:

Case 1: *Q1 = y(0)*, stepsize: 6,6,6,8,8,8,10,10,10..…..

Case 2: *Q1=y(16T),* stepsize: 6,6,6,10,10,10,……

The results are shown in Figure 15 and 16, respectively.

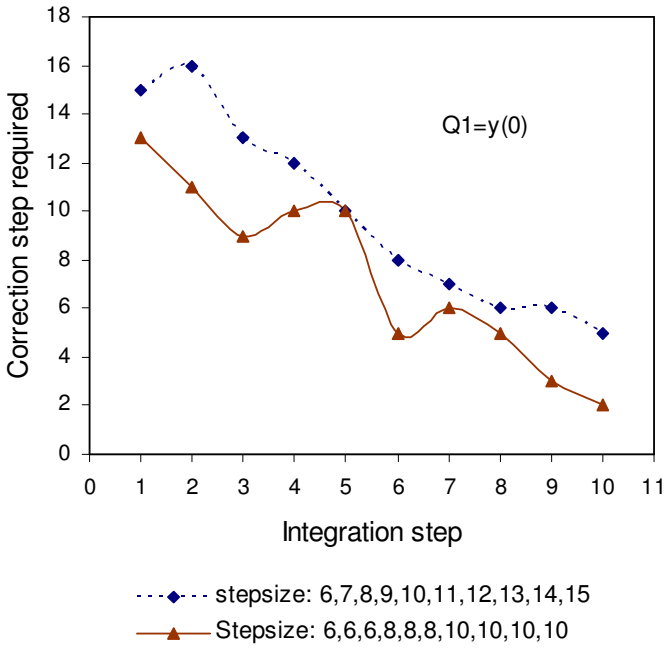


Figure 15 Outer integrator performance under variable stepsize Q1=y(0)

Both cases indicate the increasing work load where the stepsize change occurred. In general, a variable stepsize produces slow convergence compared to the constant stepsize strategy. When iteration processes used for solving the implicit relations arisen in ODE-IVP methods, it only starts to converge rapidly after a certain number of iterations. We have observed this from all the trials in this Section. This characteristic obviously prevents us from frequently changing the stepsize, unless other iteration method is imposed to achieve faster convergence right from the beginning. There are many modifications to this method in the aim of improving global convergence property. However, it is beyond the scope of this study and will be further investigated in the future.

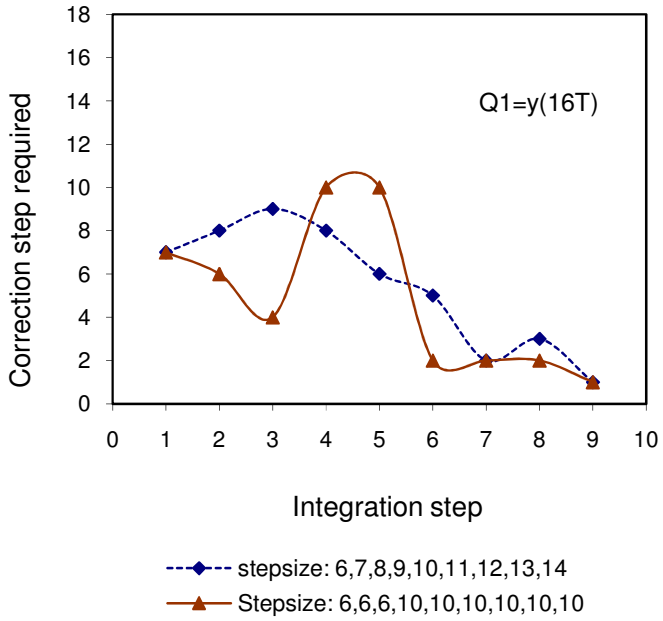


Figure 16 Outer integrator performance under variable stepsize Q1=y(16T)

6. CONCLUSIONS

Earlier determination of steady state is significantly beneficial for the evaluation of the SMB performance so that strategies can be decided ahead for improved process control. In this research, we have successfully developed a Quasi-Envelope based computation scheme for faster prediction of the steady state of SMB processes. The scheme stems from our understanding that the SMB operation can be considered as a highly oscillatory process, leading to a completely new and innovative approach for achieving the steady state. Avoiding the detailed calculation of transit equations, it provides an indirect prediction via a pseudo smooth function. The computational cost for the determination of the steady state is remarkably reduced.

Some useful conclusions can be drawn from this investigation. Firstly, if the starting state of the Quasi-Envelope is chosen the same as the original function, the multivalue algorithm would require similar computational effort (total number of correction steps) to achieve the steady state prediction, regardless of the integration stepsize. Secondly, if using constant stepsize, activating QE after the initial start-up period (1-2 cycles) is helpful when quasi-envelope displays steep change at that stage. Thirdly, a variable stepsize produces slow convergence compared to the constant stepsize strategy, thus increasing the workload where the stepsize change is occurring. It requires other iteration methods to be imposed to the proposed algorithm. Potential applications can be seen to other industrial processes with inherent cyclic behaviour.

NOMENCLATURE

$C_{i,j}$: liquid phase concentrations of component *i* in column *j*

$C_{i,j}^{in}$, $C_{i,j}^{out}$:concentration of component *i* at the outlet or the inlet of column *j*

D: column diameter

$D_{ax,j}$: axial dispersion coefficient of the fluid phase in column *j*

g_n: first-order derivative of z obtained in the prediction step

$g(z_n)$: calculated first-order derivative of z, defined as $g(z_n) = \dfrac{z(t_n + T) - z(t_n)}{T}$

H: step size for integration

J : wavelet resolution level

$k_{eff,j}$: effective mass transfer coefficient in column *j*

L: column length

M : interpolation degree of boundary treatment in wavelet-collocation method

Q_D : desorbent flow rate

Q_E : extract flow rate

Q_F : feed flow rate

Q_R : raffinate flow rate

$Q1$: the initial state of z

Q_I , Q_{II} , Q_{III} , Q_{IV} : volumetric flow rate through the corresponding sections

$q*$: equilibrium concentration in interface between two phases

$q_{i,j}$: solid phase concentration of component *i* in column *j*

R_n : stepsize ratio for the consecutive integration step

T : duration of one period

t : time coordinates

t_s : switching time

u_j : interstitial velocity in column *j*

x : axial coordinate

y : periodic or near periodic function

z : Quasi-Envelope function

$\alpha; \beta$: coefficients in multistep integration formula

ε_b : void porosity of the mobile phase

$\mathbf{\Phi}$: Jacobian matrix for the Newton algorithm

REFERENCES

1. Beste, Y.A., Lisso, M., Wozny, G., Arlt, W., 2000. Optimisation of simulated moving bed plants with low efficient stationary phases: separation of fructose and Glucose. Journal of Chromatography A 868, 169-188.
2. Biegler, L.T., Jiang L., Fox, V.G., 2004. Recent advances in simulation and optimal design of pressure swing adsorption systems. Separation and Purification Reviews 33(1), 1-39.
3. Ching, C.B., 1997. Parabolic intraparticle concentration profile assumption in modelling and simulation of nonlinear simulated moving-bed separation processes. Chemical Engineering Science 53(6), 1311-1315.
4. Chu, K.H., Hashim, M.A., 1995. Simulated countercurrent adsorption processes: a comparison of modelling strategies. The Chemical Engineering Journal 56, 59-65.
5. Davis, M.E., 1984. Numerical Methods & Modelling for Chemical Engineers. John Wiley & Sons, Inc., New York.
6. Dünnebier, G., Fricke, J., Klatt, K-U., 2000. Optimal design and operation of simulated moving bed chromatographic reactors. Industrial and Engineering Chemistry Research 39, 2290-2304.
7. Gear, C.W., 1971. Numerical Initial value problems in ordinary differential equations. Prentice-Hall, Inc., New Jersey.
8. Hashimoto, K., Adachi, S., Noujima, H., Maruyama, A., 1983. Models for the separation of glucose/fructose mixture using a simulated moving-bed adsorber. Journal of Chemical Engineering in Japan 16, 400-406.
9. Jiang, L., Biegler, L., Fox, V., 2003. Simulation and optimization of pressure-swing adsorption systems for air separation. American Institute of Chemical Engineering Journal 49, 1140-1157.
10. Kloppenburg, E., Gilles, E.D., 1999. A new concept for operating simulated moving-bed processes. Chemical Engineering Technology 22, 813-817.
11. Lübke, R., Seidel-Morgenstern, A., Tobiska, L., 2007. Numerical method for accelerated calculation of cyclic steady state of ModiCon-SMB-processes. Computers and Chemical Engineering 31, 258-267.
12. Minceva, M., Pais, L.S., Rodrigues, A.E., 2003. Cyclic steady state of simulated moving bed processes for enantiomers separation. Chemical Engineering and Processing 42, 93-104.
13. Nilchan, S., Pantelides, C., 1998. On the optimisation of periodic adsorption processes. Adsorption 4, 113-147.
14. Pais, L.S., Loureiro, J.M., Rodrigures, A.E., 1998. Modelling strategies for enantiomers separation by SMB Chromatography. American Institute of Chemical Engineering Journal 44, 561-569.

15. Petzold, L.R., (1981). An efficient numerical method for highly oscillatory ordinary differential equations. SIAM J Numerical Analysis 18, 455-479.

16. Platte, F., Kuxmin, D., Fredebeul, C., Turek, S., 2005. Novel simulation approaches for cyclic-steady-state fixed-bed processes exhibiting sharp fronts and shocks. In de Bruin, M., Mache, D., Szabados, J. (Eds.), Trends and applications in constructive approximations. Vol. 151 of international series of numerical mathematics (pp. 207-223). Birkhäuser Verlag, Basel, 2005.

17. Ruthven, D.M., 1984. Principles of Adsorption and Adsorption Process. John Wiley & Sons, New York.

18. Storti, G., Masi, M., Paludetto, R., Morbidelli, M., Carrà, S., 1988. Adsorption separation processes: countercurrent and simulated countercurrent operations. Computers and Chemical Engineering 12, 475-482.

19. Strube, J., Schmidt-Traub, H., 1998. Dynamic simulation of simulated-moving-bed chromatographic processes. Computers and Chemical Engineering 22, 1309-1317.

20. Toftegard, B., Jorgensen, S.B., 1989. An integration method for dynamic simulation of cycled processes. Computers and Chemical Engineering 13, 927-930.

21. Yao, H.M., Tian, Y.C., Tadé, M.O., 2008. Using Wavelets for Solving SMB Separation Process Models. Industrial & Chemical Engineering Research 47, 5585-5593.